МИНИСТЕРСТВО НАУКИ И ОБАЗОВАНИЯ

Федеральное государственное бюджетное образовательное учреждение

Высшего профессионального образования

**«Вятский государственный университет»**

**(ФГБОУ ВПО «ВятГУ»)**

Факультет автоматики и вычислительной техники

Кафедра электронных вычислительных машин

Микропрограммная реализация ЭВМ

Отчет по лабораторной работе №4 дисциплины

«Организация ЭВМ и систем»

Выполнил студент группы ИВТб-31\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_/Седов М.Д./

Проверил преподаватель \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_/Клюкин В.Л./

Киров 2019

1. Задание на лабораторную работу

Подсчитать число одинаковых цифр, расположенных в одноименных разрядах xi и yi 8-разрядных двоичных чисел X и Y.

1. Определение архитектуры и программирование

*Форматы данных.* Данные в примере являются целыми числами, изменяющимися в пределах от 0 до 32767. В этом случае любое число можно представить 16-разрядным двоичным кодом, старший разряд которого определяет знак числа.

*Программно – доступные регистры.* Программно – доступными регистрами МЭ, система команд которой состоит из одноадресных команд, можно считать: аккумулятор – AC, программный счётчик – PC и регистр признаков – RP, содержащий в простейшем случае разряды двух признаков: нуля (PZ) и знака (PS).

*Система команд.* Разработка системы команд предполагает определение и кодирование операций, способов адресаций, модификаций и форматов данных. Граф – схема алгоритма деления чисел изображена на рисунке 1.

Для системы команд, состоящей из одноадресных команд с прямой адресацией, достаточно одного формата команды, содержащего два поля: кода операции К и адреса А. Система команд МЭ представлена в таблице 1, где M[A] – ячейка памяти с адресом А, знак “+” в описании признаков означает, что устанавливается новое значение признака по результату выполнения команды, а знак “-“ свидетельствует о сохранении старого значения признака.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Наименование | Мнемоника | Описание | Признаки | |
| PZ | PS |
| ОЧИСТКА | CLM A | M[A]:=00…0, PC:=PC+1 | - | - |
| ЗАПИСЬ АС | MOV A | M[A]:=AC; PC:=PC+1 | - | - |
| ИНКРЕМЕНТ | INC A | M[A]:=M[A]+1; PC:=PC+1 | - | - |
| СДВИГ ВПРАВО | SHR A | AC:=AC>>1; PC:=PC+1 | + | + |
| НЕ ИСКЛЮЧАЮЩЕЕ ИЛИ | NXOR A | AC:=^(AC xor M[A]) | - | - |
| ЗАГРУЗКА АС | LDA A | AC:=M[A]; PC:=PC+1 | + | + |
| НЕТ ОПЕРАЦИИ | NOP | PC:=PC+1 | - | - |
| ПЕРЕХОД | BR A | PC:=A | - | - |
| ПЕРЕХОД,  ЕСЛИ НУЛЬ | BEQ A | Если PZ=1, то PC:=A,  иначе PC:=PC+1; | - | - |
| ОСТАНОВ | HLT A | PC:=A, останов | - | - |

Таблица 1 – Система команд

Команды МЭ в зависимости от выполнения операций можно разделить на три группы.

*Арифметико – логические команды:* умножение, очистка, инкремент, декремент. Операция умножения является двухместной и может быть представлена общим выражением: c:=a\*b, где а – источник первого, b – источник второго операнда; c – приемник результата, \* - символ операции. В общем случае двухместные операции требуют введения в команду трех указателей: a,b, c. Число указателей может быть сокращено до двух, если результат операции помещается на место одного из операндов: a:=a\*b или b:=a\*b. Для рассматриваемых одноадресных команд a, b ϵ {AC, M[a]}.

Операции очистки, инкремента и декремента являются одноместными: a:=\*(a). Для рассматриваемых команд a ϵ {AC, M[a]}.

*Команды – пересылки.* Эти команды представлены в системе команд МЭ командой загрузки аккумулятора и командой записи содержимого аккумулятора в память. Операции пересылки могут быть описаны общим выражением: a:=b; a, b ϵ {R[N] M[a]}, где R[N] – один из внутренних регистров МЭ с номером (идентификатором) N.

*Команды управления:* переход по нулю, переход по минусу, переход безусловный, останов с загрузкой программного счётчика.

Для рассматриваемых одноадресных команд операции перехода можно представить следующим выражением: Nj+1=F(Nj, a, p), где Nj – номер (адрес) текущей; Nj+1 – следующей команды; a – указатель; p – условие перехода; F(Nj, a, p) – функция формирования номера (адреса) следующей команды. Тогда для операции условного перехода по нулю (знаку) Nj, Nj+1 ϵ {PC}, a ϵ {A}, p ϵ {PZ} (p ϵ {PS}). А в операциях безусловного перехода PC:=A.

Программа подсчета равных чисел в разрядах двух чисел c использованием символики ассемблера приведена на рисунке 1.

|  |  |  |
| --- | --- | --- |
|  | CLM AP | Очистка ЯП для P |
|  | LDA AX | Загрузка в AC X |
|  | NXOR AY | Исключ или с Y |
| m1 | BEQ m2 | Если PZ=1(Y=0), то переход на метку m2 |
|  | SHR AC | Сдвиг АС вправо |
|  | INC AP | Увеличение на единицу результата P |
|  | BR m1 | Переход на метку m1 |
| m2 | HLT SA | Загрузка PC и останов |

Рисунок 1 – Программа подсчета равных чисел в разрядах двух чисел

В программе приняты следующие обозначения:

AX – адрес ячейки памяти (ЯП), в которой находится первое число;

AY – адрес ячейки памяти (ЯП), в которой находится второе число;

AP – адрес ячейки памяти (ЯП), в которой находится количество одинаковых чисел;

SA – начальный адрес программы.

Наименования, мнемонические обозначения и коды операция приведены в табл. 2.

Табл. 2 – Коды операций

|  |  |  |
| --- | --- | --- |
| Наименование | Мнемоника | Код операции |
| ОЧИСТКА | CLM | 00000001 |
| ЗАПИСЬ АС | MOV | 00000010 |
| ИНКРЕМЕНТ | INC | 00000011 |
| ЗАГРУЗКА АС | LDA | 00000100 |
| НЕТ ОПЕРАЦИИ | NOP | 00000101 |
| ПЕРЕХОД | BR | 00000110 |
| ПЕРЕХОД, ЕСЛИ НУЛЬ | BEQ | 00000111 |
| СДВИГ ВПРАВО | SHR | 00001000 |
| НЕ ИСКЛЮЧАЮЩЕЕ ИЛИ | NXOR | 00001001 |
| ОСТАНОВ | HLT | 00000000 |

Распределение памяти программ и данных приведено в табл. 3.

Табл. 3 – Распределение памяти программ и данных

|  |  |  |  |
| --- | --- | --- | --- |
| Адрес | Код | Мнемоника | Комментарий |
| 00 | 0005 | SA | Начальный адрес программы |
| 01 |  | X | Первое число |
| 02 |  | Y | Второе число |
| 03 |  | P | Количество одинаковых цифр |
| 04 |  |  | Свободная ячейка памяти (ЯП) |
| 05 | 0003 | CLM AP | Очистка ЯП для P |
| 06 | 0101 | LDA AX | Загрузка в AC X |
| 07 | 0202 | NXOR AY | Исключ или с Y |
| 08 | 040С | BEQ m2 | Если PZ=1(Y=0), то переход на метку m2 |
| 09 | 0308 | SHR AC | Сдвиг АС вправо |
| 0A | 0603 | INC AP | Увеличение на единицу результата P |
| 0B | 0708 | BR m1 | Переход на метку m1 |
| 0C | 0505 | HLT SA | Загрузка PC и останов |

Каждому программно – доступному регистру (AC, RP, PC) ЭВМ ставится в соответствие один из внутренних регистров микропроцессора. Программно – недоступные регистры выделяются с учетом функциональных возможностей микропроцессора и максимальным числом одновременно хранимых промежуточных результатов выполнения команд. Распределение регистров показано на рис. 2. Функции триггеров FS и FZ ЭВМ выполняют разряды S и Z регистра состояний операционного устройства микропроцессора.

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
|  | РЗУ (R0-R7) | | |  |  | РЗУ (R8-R15) |
| 0: | АС | | |  | 8: |  |
| 1: |  | | |  | 9: |  |
| 2: |  | | |  | 10: |  |
| 3: |  | | |  | 11: |  |
| 4: |  | | |  | 12: |  |
| 5: |  | | |  | 13: | Регистр для данных ЗУ |
| 6: | РС | | |  | 14: | Регистр константы |
| 7: | PS | RP | PZ |  | 15: | Счетчик адреса ЗУ RK[A] |

Рис. 2 – Распределение регистров

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| № | МИ | РЗУ | | Упр. АЛУ | | | Упр. ОЗУ | | | Шина | МИ | Упр. усл. | | | Упр. УУ | | |
| N | I8-0 | A | B | C0 |  | SC |  |  |  | D11-0 | I3-0 | A | U |  | C0 |  |  |
| 00 | 571 | E | E | 0 | 0 | 00 | 1 | 1 | 1 | 006 | C | 00 | 0 | 0 | 1 | 1 | 0 |
| 01 | 533 | 0 | E | 0 | 0 | 00 | 1 | 1 | 1 | 001 | 9 | 00 | 0 | 0 | 1 | 1 | 0 |
| 02 | 143 | 0 | 6 | 0 | 0 | 00 | 1 | 1 | 0 | 000 | E | 00 | 0 | 0 | 1 | 1 | 0 |
| 03 | 337 | 0 | 6 | 0 | 1 | 00 | 0 | 1 | 1 | 000 | E | 00 | 0 | 0 | 1 | 1 | 0 |
| 04 | 203 | 6 | 6 | 1 | 0 | 00 | 1 | 1 | 0 | 000 | E | 00 | 0 | 0 | 1 | 1 | 0 |
| 05 | 245 | E | F | 0 | 1 | 00 | 0 | 1 | 1 | 000 | 2 | 00 | 0 | 0 | 1 | 1 | 0 |
| 06 | 343 | 0 | 7 | 0 | 1 | 00 | 1 | 1 | 1 | 000 | E | 00 | 0 | 0 | 1 | 1 | 0 |
| 07 | 133 | 0 | 0 | 0 | 1 | 00 | 1 | 1 | 1 | 004 | 3 | 0 | 0 | 0 | 1 | 1 | 0 |
| 08 | 303 | 0 | 7 | 1 | 1 | 00 | 1 | 1 | 1 | 004 | 3 | 00 | 0 | 1 | 1 | 1 | 0 |
| 09 | … | . | . | . | . | .. | . | . | . | … | . | .. | . | . | 1 | 1 | 0 |
| 0A | 133 | 0 | F | 0 | 0 | 00 | 1 | 1 | 0 | 000 | E | 00 | 0 | 0 | 1 | 1 | 0 |
| 0B | 143 | 0 | F | 0 | 0 | 00 | 0 | 0 | 1 | 004 | 3 | 00 | 0 | 1 | 1 | 1 | 0 |
| 0C | 133 | 0 | F | 0 | 0 | 00 | 1 | 1 | 0 | 000 | E | 00 | 0 | 0 | 1 | 1 | 0 |
| 0D | 337 | 0 | 0 | 0 | 1 | 00 | 0 | 1 | 1 | 004 | 3 | 00 | 0 | 1 | 1 | 1 | 0 |
| 0E | 133 | 0 | F | 0 | 0 | 00 | 1 | 1 | 0 | 000 | E | 00 | 0 | 0 | 1 | 1 | 0 |
| 0F | 375 | 0 | 0 | 0 | 1 | 00 | 0 | 1 | 1 | 004 | 3 | 00 | 0 | 1 | 1 | 1 | 0 |
| 10 | 434 | 0 | 0 | 0 | 1 | 00 | 1 | 1 | 1 | 006 | 3 | 00 | 0 | 1 | 1 | 1 | 0 |
| 11 | 113 | 0 | 7 | 0 | 1 | 00 | 1 | 1 | 1 | 004 | 3 | 00 | 0 | 0 | 1 | 1 | 0 |
| 12 | 334 | F | 6 | 0 | 1 | 00 | 1 | 1 | 1 | 004 | 3 | 00 | 0 | 1 | 1 | 1 | 0 |
| 13 | 334 | F | 6 | 0 | 1 | 00 | 1 | 1 | 1 | 000 | 3 | 00 | 0 | 1 | 1 | 1 | 0 |
| 14 | 133 | 0 | F | 0 | 0 | 00 | 1 | 1 | 0 | 000 | E | 00 | 0 | 0 | 1 | 1 | 0 |
| 15 | 337 | 0 | D | 0 | 1 | 00 | 0 | 1 | 1 | 000 | E | 00 | 0 | 0 | 1 | 1 | 0 |
| 16 | 303 | 0 | D | 1 | 0 | 00 | 0 | 0 | 1 | 004 | 3 | 00 | 0 | 1 | 1 | 1 | 0 |

Таблица 4 – Микропрограмма командного цикла